IN THE SPECIFICATION

Please amend the specification as follows:

The paragraph beginning at page 3, line 18 is amended as follows:

Figure 1 is a schematic block diagram of an electronic system 100 in accordance with an embodiment of the present invention. The electronic system 100 includes at least two semiconductor chips 102a and 102b, devices or circuits that communicate with one another. The electronic system 100 may include additional semiconductor chips, devices or circuits 102c and 102d. The electronic system 100 may also be formed as a single chip and the circuits or devices 102 may each be an on-chip silicon module. For example, semiconductor chips 102a and 102c may be processors, such as central processing units (CPUs), digital signal processors (DSPs) ~~for~~ or the like, and the semiconductor chips 102b and 102d may be memory devices, peripheral equipment ~~for~~ or the like. The chips 102a and 102c are coupled to an internal bus 110, such as a processor bus or a peripheral bus, and the internal bus 110 is coupled to a first communications interface 112 or multiple channel interface for communications between the chips 102a, 102b ,102c and 102d. The first communications interface 112 may be a communications interface between any different types of chips or may be a broadband–to–multimedia (BB-MM) interface for communications between a multimedia processor and a baseband chip. The first communications interface 112 is electrically coupled to a second communications interface 114 by a plurality of outgoing or outbound links or pin connections 116 and a plurality of incoming or inbound links or pin connections 118. The outbound links or pins 116 from the first communications interface 112 are the inbound links to the second communications interface 114 and the inbound links 118 to the first communications interface 112 are the outbound links from the second communications interface 114. The second communications interface 114 is electrically coupled to an internal bus 120 and the bus 120 is electrically coupled to be semiconductor chips 102b and 102d, if there are more than one chip 102 coupled to the second communications interface 114.

The paragraph beginning at page 5, line 15 is amended as follows:

In accordance with an ~~embodiments~~ embodiment of the present invention, the transmit control block 308 may include a link controller 318 and a channel arbiter 320. The channel arbiter 320 determines which of the plurality of transmit channels 306 is to be activated or selected next to transmit data. As described in more detail below this could be one of a plurality of data channels, a virtual general-purpose input/output (GPIO) channel 307 or a message flow control (MFC) channel. The channel arbiter 320 uses as inputs the transmit channels 306, if any, that may be in a "wait" state and therefore cannot transmit data for some reason, the channel number of the currently activated transmit channel 306, and information from each of the transmit channels 306 or transmit FIFOs 314 indicating if they contain any data to be sent. The channel arbiter 320 outputs the channel number of the next transmit channel 306 or FIFO 314 to be activated for transmitting data.

The paragraph beginning at page 6, line 6 is amended as follows:

The receive control block 312 may include a state machine 322 that stores the channel number of the currently active receive channel ~~306~~ 310 or FIFO 316, the number of data bits in the current byte that have been transmitted and the data bits themselves in the current byte that have already been received. Using this information the state machine 322 ~~will write~~ writes each byte to the correct receive channel 310 or receive FIFO 316 after the state machine 322 ~~has receive~~ receives a complete byte of data. The state machine 322 may also be implemented in software.

The paragraph beginning at page 6, line 13 is amended as follows:

In accordance with an embodiment of the present invention, the communications interface 112 may include a power management unit 324. The power management unit 324 may be contained within the bus interface 300 or may be external to the bus interface 300. The power management unit 324 may be coupled to the plurality of transmit channels 306, plurality of receive channels 310, the transmit and receive control ~~block~~ blocks 308 and 312, and the semiconductor chips 102 (Figures 1 and 2). The power management unit 324 facilitates placing the components of the system 200 into an idle state or sleep state or mode to conserve energy as

described in more detail below. Before entering a sleep mode, the components may perform software handshaking. For example, the components may cause a sleep request message to be sent, receive an okay response and then enter the sleep mode. Sending and receiving these messages can be implemented by using any channel 306 or 307 and an associated control ~~registers~~ register 302. Before sending a request–to–sleep message, the requesting chips 102 should transmit all of its data or messages in any transmit channels 306 or FIFOs 314 to stop outbound activity. Before responding with the okay to sleep message, the receiving chip 102 should receive all messages directed to it or drain any receive channels 310 or FIFOs 316 containing messages for the receiving chip 102 and terminate all receive activity. Once the requesting chips 102 receive the okay to sleep response, it can safely enter the sleep mode.

The paragraph beginning at page 7, line 6 is amended as follows:

Figure 4 is a schematic block diagram of the communications interface 112 illustrating examples of a transmit control block 308 and a receive control block 312 in accordance with another embodiment of the present invention. The transmit control block 308 may include a multiplexer or mux 400 coupled to the plurality of transmit channels 306. The multiplexer 400 is coupled to a parallel in serial out (PISO) converter 402. The PISO converter 402 is coupled to a channel selector 404 and a control logic circuit 406. The channel selector 404 and the control logic circuit 406 are each coupled to a channel register 408 that is also coupled to the PISO converter 402. The channel selector 404 and the control logic circuit 406 are also connected to a channel configuration register 410 and a channel status register 412. The channel configuration register 410 and the channel status register 412 may be included as part of the transmit control registers 302 (Figure 3) and contained in the bus interface 300. A channel configuration register 410 may be associated with each transmit channel 306 and each receive channel 310 and provides information about the specific transmit channel 306 or receive channel 310 such as the type of service selected (DMA, interrupt), the threshold level, the type message flow control and the like. A channel status register 412 may also be associated with each transmit channel 306 and each receive channel 310 and ~~provides~~ provide information about the channel 306 or 310 such as whether the channel 306 or 310 or FIFO 314 or 316 is in a "Wait" state, is empty or full and the degree or amount of fullness or emptiness or if there is any data in the FIFO 314 or 316.

The paragraph beginning at page 9, line 22 is amended as follows:

The receive control logic circuit 422 may transmit a "Wait" signal 423 over a "Wait" link 424 to the transmit control logic circuit 406 of the transmit control block 308 if the configuration and status data from the receive channel configuration register 410 and the receive channel status register 412 indicate that the new receive channel 310 or FIFO 316 is full, disabled or otherwise cannot receive data. The control logic circuit 422 will also generate and send a "WRITE STROBE" signal 425 to the demultiplexer 418 when a whole or complete byte of data has been received by the SIPO converter 416. The demultiplexer 418 selects the proper receive channel 310 in response to the channel number from the channel register 420. Parallelized data from the SIPO converter 416 ~~will then be~~ is then written into the selected one of the receive FIFOs 316 in response to the "WRITE STROBE" signal 425.

The paragraph beginning at page 11, line 23 is amended as follows:

Referring back to Figure 5, the interface interrupt ID register 512 may be coupled to the chip 102 and to the transmit control block 308. Figure 8 shows an example of a bit layout and bit definitions for the interface interrupt ID register 512. Interrupts can be generated when a transmit FIFO 314 or a receive FIFO 316 reaches its threshold value set by its corresponding channel configuration ~~registers~~ register 410, when an end of message (EOM) signal is ~~receive~~ received, or when a DMA descriptor chain is reached before the end of a message is read. Generating an interrupt on an early DMA end of channel is necessary to ~~inform~~ signal a processor type chip 102 of improper DMA programming. As shown in Figure 8, each interface interrupt type has a bit associated with it in the interface interrupt ID register 512. When an interface interrupt occurs, the corresponding bit is set in the interface interrupt ID register 512.

The paragraph beginning at page 12, line 4 is amended as follows:

Referring back to Figure 5, the transmit frequency select register 514, the wait count register 516, the clock stop time register 518 and the interface width register 520 may each be coupled between the chip 102 and the transmit control block 308. The transmit frequency select register 514 selects the clock speed of the outbound link 116. The wait count register 516

determines the time (in transmit clock cycles) that the transmit control block 308 ~~will wait~~ waits before retrying a transmit to a receive channel 310 that ~~sent~~ sends a wait signal 423 (Figure 4). Each transmit channel 306 has an independent wait count register 516 that counts the time after a wait signal 423 is received before a retransmission to the receive channel 310 that caused the wait signal 423 to be sent. The clock stop time register 518 determines the time (in transmit clock cycles) that a clock signal ~~will stop~~ stops transitioning after the outbound link 116 become idle. It should be noted that a clock signal may be generated by the interfaces 112 and 114 of the present invention only when needed. The interface width register 520 specifies the width or number of data links 116 that the first communications interface 112 ~~will transmit~~ transmits over simultaneously to the second communications interface 114.

The paragraph beginning at page 16, line 15 is amended as follows:

As previously mentioned, the communications interfaces 112 and 114 may support different interface widths. The appropriate bit in the interface width register 520 may be set to provide the different interface widths, such as a serial width or mode (1 bit), a two-bit width or mode, a nibble width or mode (4-bits) and so forth. In the example shown in Figure ~~13 the~~ 13, the signals 1302, 1304, 1306 and 1308 are in the nibble width or mode. Accordingly, four data links or pins 1108 in Figure 11, DAT(0), DAT(1), DAT(2) and DAT(3), may be used to transmit a message in nibble mode.

The paragraph beginning at page 17, line 15 is amended as follows:

While receiving data, the receive FIFO 316 (Figure 3) can become full which ~~would prevent~~ prevents the receive FIFO 316 from accepting new data. One example of a flow control method to notify the source communications interface 112 of this condition may be referred to as direct flow control (DFC) and another example of a flow control method for a FIFO full condition may be referred to as message flow control (MFC). Both methods temporarily disable data transfers by putting the active transmit channel 306 or FIFO 314 in a "wait" state. When the active transmit channel 306 is in a wait state, the source communications interface 112 cannot send any data through that channel 306. Any attempt to transmit data will be ignored. Either or both flow control methods can be used by the communications interfaces 112 and 114.

The paragraph beginning at page 17, line 25 is amended as follows:

Referring back to Figure 11, in the direct flow control method, the target interface 114 ~~will assert~~ asserts a "wait" signal 1308 over the wait link or pin 1106 to the source interface 112, if the active receive channel 310 or FIFO 316 is disabled, invalid or full. The "wait" signal 1308 ~~will also be~~ is also sent after a reset and while the data link 1108 is idle, i.e., there is no data or messages being transmitted. The source interface 112 will sample the "wait" signal 1308 on each CLK pulse 1302 of the CLK link 1102 while the active data channel 1-7 is in a wait state for as long as the "wait" signal 1308 is being asserted. When the "wait" signal 1308 goes low or is no longer asserted, data transmission can resume. Another data channel 1-7 may be activated while the currently active channel 1-7 is in a wait state by transmitting the STB signal 1306 on the strobe link 1104 and transmitting a new data channel number 1-7 on a corresponding data link 1108.